A Tool for Visualizing the Topology of Three-Dimensional Vector Fields

A. Globus, Computer Sciences Corporation
C. Levit, NASA Ames Research Center
T. Lasinski, NASA Ames Research Center

Report RNR-91-017, April 1991

National Aeronautics and Space Administration

Ames Research Center
Moffett Field, California 94035

ARC 275a (Feb 81)

A Tool for Visualizing the Topology of Three-Dimensional Vector Fields

A. Globus, Computer Sciences Corporation
C. Levit, NASA Ames Research Center
T. Lasinski, NASA Ames Research Center

Report RNR-91-017, April 1991

To appear in Visualization '91, October 22-25, 1991, San Diego, CA.

This work was supported in part by NASA Contract NAS 2-12961 to Computer Scineces Corporation for the Numerical Aerodynamic Simulation Systems Division at NASA Ames Research Center.

Copies of this report are available from:
NAS Applied Research Office
Mail Stop T045-1
NASA Ames Research Center
Moffett Field, CA 94035
(415) 604-4332

A Tool for Visualizing the Topology of Three-Dimensional Vector Fields

A. Globus, Computer Sciences Corporation
C. Levit, NASA Ames Research Center
T. Lasinski, NASA Ames Research Center

Abstract

We describe a software system, TOPO, that numerically analyzes and graphically displays topological aspects of a three dimensional vector field **v** to produce a single, relatively simple picture that characterizes **v**. The topology of **v** that we consider consists of its critical points (where $\mathbf{v} = 0$), their invariant manifolds, and the integral curves connecting these invariant manifolds. Many of the interesting features of **v** are associated with its critical points. The field in the neighborhood of each critical point is approximated by the Taylor expansion. The coefficients of the first non-zero term of the Taylor expansion around a critical point are the 3x3 matrix $\nabla\mathbf{v}$. Critical points are classified by examining $\nabla\mathbf{v}$'s eigenvalues. The eigenvectors of $\nabla\mathbf{v}$ span the invariant manifolds of the linearized field around a critical point. Curves integrated from initial points on the eigenvectors a small distance from a critical point connect with other critical points (or the boundary) to complete the topology. In addition, one class of critical surfaces important in computational fluid dynamics is analyzed.

TOPO is implemented as a module in the FAST [1] visualization environment. It operates on curvilinear, structured grids, including large multi-zone grids. We have used TOPO to visualize a number of CFD data sets. The results agree well with other topology software and hand generated topologies. TOPO has proved useful in finding surface topology, flow attachment and separation points, vortex cores, scalar field local extrema, and generally interesting regions of **v**. We believe there may be other interesting applications yet to be discovered. This paper contains most of the information needed for a good programmer to code a topology module in another environment.

Introduction

Large, complex vector fields are difficult to visualize. One simple technique chooses a set of points in the field and draws arrows indicating the magnitude and direction of **v** at each point. Unfortunately, this usually results in a display that is either terribly cluttered or limited to a small subset of the data. If the points are chosen from some simple shape like a plane, the ends of the vectors can be connected to form a deformation surface -- but again only a subset of data is visualized. One can calculate the vector magnitude and use scalar visualization techniques but directional information is lost. Interactive control of initial positions for integral curves (a.k.a. "particle traces") [2] and surfaces [3] may be used to explore **v**, but choosing appropriate initial positions is hardly straightforward.

A simple picture completely characterizing **v** is the ideal. Studying such a picture should give one a clear and complete understanding of the important characteristics of **v**. Vector field topology visualization has the potential to go a long way in this direction; our software takes one small step.

Informally, vector field topology consists of the key points, curves and surfaces that, taken together, characterize all integral manifolds in **v**. Integral manifolds include particle traces, streamlines,

The i,j,k indexes into this array are equivalent to integer part of the (ξ, η, ζ) co-ordinates. We use computation space for calculations since they simplify many operations; e.g., differencing.

$\mathbf{v} = (u, v, w)$ is defined as:

$$\begin{aligned} u(x, y, z) &= \frac{dx}{dt} \\ v(x, y, z) &= \frac{dy}{dt} \\ w(x, y, z) &= \frac{dz}{dt} \end{aligned} \qquad (2)$$

or

$$v_i = \frac{dx_i}{dt}$$

Where, in this case, (x,y,z) are the lagrangian coordinates of an element "moving" with the vector field and t is time in velocity fields and an arbitrary parameter in others. $\mathbf{v}(x, y, z)$ may be converted to computation space $\mathbf{v}'(\xi, \eta, \zeta)$ using the chain rule:

$$\begin{aligned} u'(\xi, \eta, \zeta) &= \frac{d\xi}{dt} = \frac{\partial \xi dx}{\partial x dt} + \frac{\partial \xi dy}{\partial y dt} + \frac{\partial \xi dz}{\partial z dt} \\ v'(\xi, \eta, \zeta) &= \frac{d\eta}{dt} = \frac{\partial \eta dx}{\partial x dt} + \frac{\partial \eta dy}{\partial y dt} + \frac{\partial \eta dz}{\partial z dt} \\ w'(\xi, \eta, \zeta) &= \frac{d\zeta}{dt} = \frac{\partial \zeta dx}{\partial x dt} + \frac{\partial \zeta dy}{\partial y dt} + \frac{\partial \zeta dz}{\partial z dt} \end{aligned} \qquad (3)$$

or,

$$\mathbf{v}' = \mathbf{J}\mathbf{v}$$

Where $\mathbf{J}$ is the Jacobian matrix of the coordinate transformation. $\mathbf{J}$ and $\mathbf{J}^{-1}$ are used to convert from physical space to computation space and back.

Critical points exist where the components of $\mathbf{v}$ all simultaneously vanish. They are identically located in both physical and computation space since zero length vectors remain invariant under nonsingular local linear transformations such as $\mathbf{J}$ and $\mathbf{J}^{-1}$.

The Matrix $\nabla \mathbf{v}$

The Taylor series expansion of $\mathbf{v}$ about a point $\mathbf{x}^{(0)}$ is:

$$v_i = v_i^{(0)} + (x_j - x_j^{(0)}) \frac{\partial v_i}{\partial x_j} + O(\Delta x_k \Delta x_l) \qquad (4)$$

The following diagrams of critical point types are after Abrahams [11].

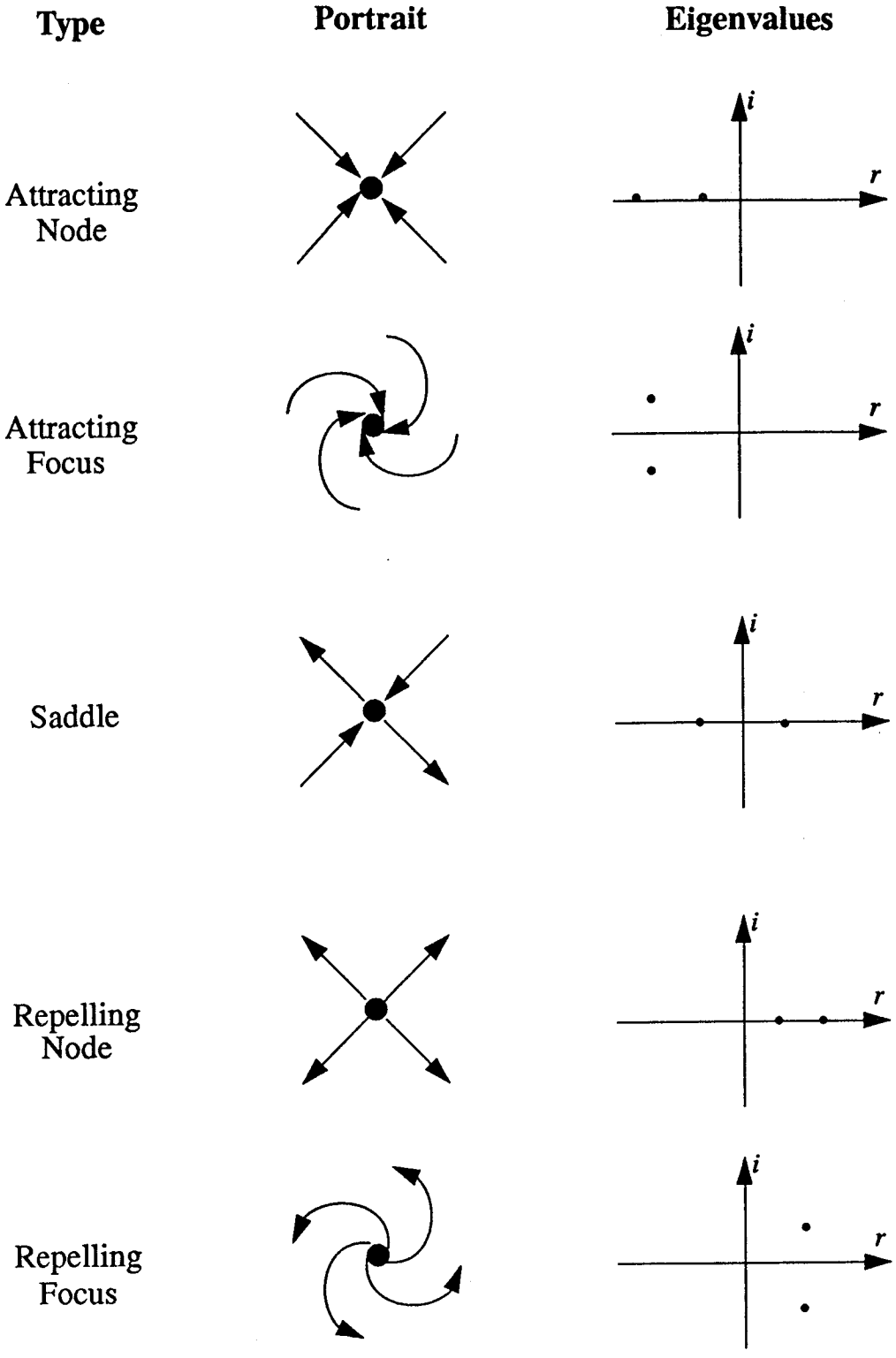

Figure 1: Classification of two dimensional critical points.

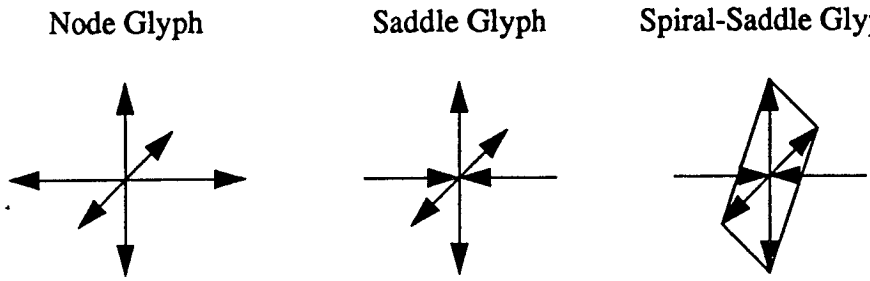

Figure 3: Critical point glyphs.

tion of **v** along an integral curve may be placed on the curve with user controlled size and separation. These arrowheads may be animated to appear to flow along the curves.

Implementation

TOPO is implemented as a module in the FAST CFD visualization environment (2). The software:

1. Transforms **v** to computation space.
2. Locates candidate grid cells that may contain critical points.
3. Locates critical points within these candidate cells.
4. Classifies each critical point using the eigenvalues of $\nabla \mathbf{v}$.
5. Integrates curves along the eigendirections of $\nabla \mathbf{v}$.
6. Displays critical points and integral curves combined with other FAST generated visual elements

Most parameters of topology generation are under user control. We have chosen defaults for these parameters that work well with the CFD data sets we have examined. These defaults are found in the Appendix B.

Transformation of v to Computational Space

The vector field is converted to computation space. This transformation is accomplished by differencing **x** with respect to $\hat{\xi}$ to generate $\mathbf{J}^{-1}$ at each grid node, inverting $\mathbf{J}^{-1}$, and transforming **v** using the resulting matrix. All of the following calculations are accomplished in computation space. Note that data points are only known at grid points so the value of **v** at other locations must be calculated using trilinear interpolation.

Finding Candidate Grid Cells

A critical point can only occur in a cell where the values of all three components of **v** pass through zero. For monotonic interpolation schemes (e.g. trilinear interpolation), this may be determined by a simple heuristic. For each component we examine the value at each cell vertex. If both negative and positive values exist, that component must change sign (and hence pass through zero) somewhere within the cell. This is a necessary, but not sufficient condition for a critical point to exist within the cell. It is not sufficient because the surfaces within a cell where the componentwise zero crossings exist might not intersect. See Figure 4.

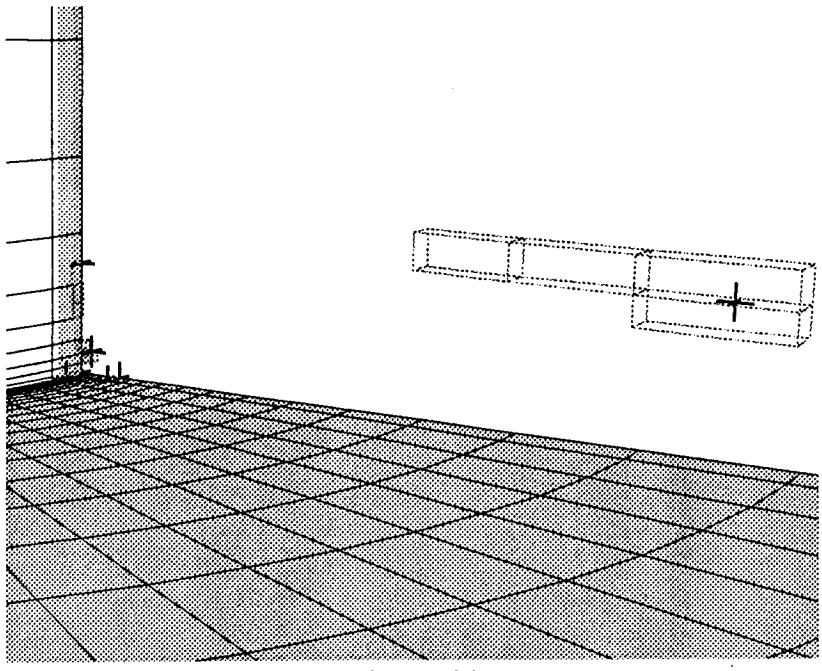

Figure 5: Critical point positions.

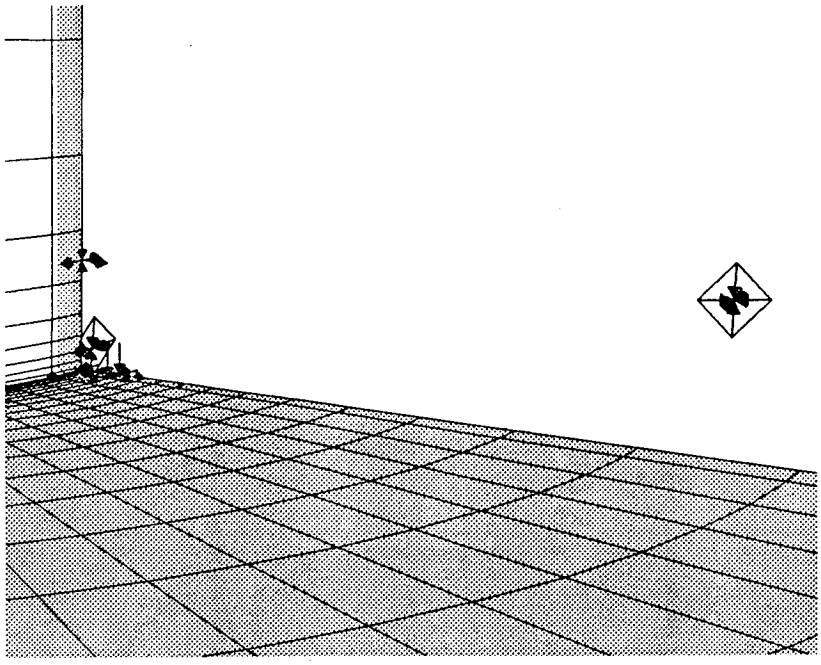

Figure 6: Critical point glyphs.

Integrating Curves

Integral curves may be computed from initial positions very close to critical points along the eigendirections of $\nabla \mathbf{v}$. These curves are integrated forward or backward depending on the sign of the eigenvalue.

Numerical Methods

Care must be taken when evaluating the eigensystems of matrices that are defective or nearly so [28]. Numerical ambiguities that arise when at or near a multiple root are handled in an ad-hoc manner that, however, seems to help. Results of subtraction are set to zero if the absolute value of the result is less than a small fraction of the absolute value of any operand.

In most places where the software tests a value for zero, there is a relative fuzz factor. Values with an absolute value less than the relative fuzz factor are assumed to equal zero. User controlled fuzz factors may be found in Appendix B.

Integration is performed in computation space using fourth order Runga-Kutta with adaptive step size error control [29]. We have found this to be necessary since the magnitude and direction of the vector field can vary dramatically near critical points. Integration is started with a user parameterized step size. The maximum number of steps, maximum stepsize, minimum and maximum step length, and the adaptation criteria in each dimension further parameterize the integration.

Differencing is accomplished using a three point stencil where possible. Where one of the values is unavailable due grid boundaries or iblanked grid points, a two point one sided difference is used.

Cell bisection is accomplished by interpolating the values at the midpoint of each cell edge, face, and at the cell center, forming eight sub-cells. This procedure is performed recursively.

Special Cases

Degenerate Critical Points

The discussion thus far has focused on "generic" or "hyperbolic" critical points. Exceptional cases can arise in several situations. Non-hyperbolic critical points occur when the real part of any eigenvalue is equal to zero. Other exceptional cases occur when defective matrices are encountered and hence eigenvectors coincide. These degenerate cases, though unstable, do occur in flows with imposed constraints such as symmetry or incompressibility. Currently, all degenerate critical points are placed in a single class by TOPO and no further analysis is attempted.

Critical Curves, Surfaces and Volumes

With the notable exception of no-slip boundaries in CFD velocity fields, TOPO makes only a minimal effort to detect, analyze, and display curves, surfaces or volumes where $\mathbf{v}$ vanishes. No attempt is made to find critical curves or surfaces unless they fall on grid cell boundaries. If two adjacent grid points have zero length vectors, we assume a critical line between them. For the most part, TOPO simply draws a line between such points. Critical surfaces on cell boundaries can be detected by examining these lines. Critical volumes can be similarly found. The cells containing such critical lines are not considered candidates and are not searched for critical points.

If a critical curve or surface exists within a grid cell, then multiple critical points may be found. TOPO limits the total number of critical points found in a single grid cell to about eight. A warning message is printed if this limit is exceeded.

No-slip Boundaries

In many CFD computations, no-slip boundary conditions are imposed on the velocity field. On

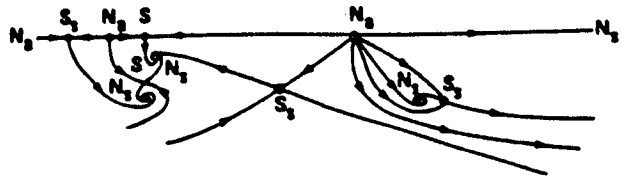

Figure 9a: Ying's hemisphere cylinder hand drawn surface flow topology.

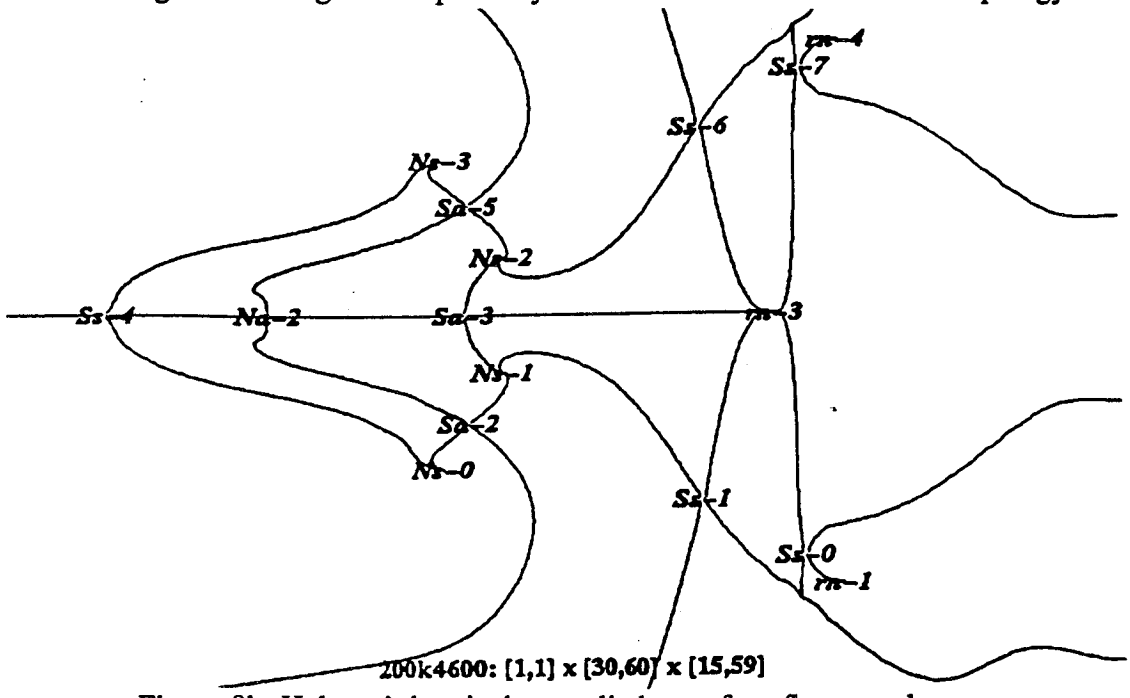

Figure 9b: Helman's hemisphere cylinder surface flow topology.

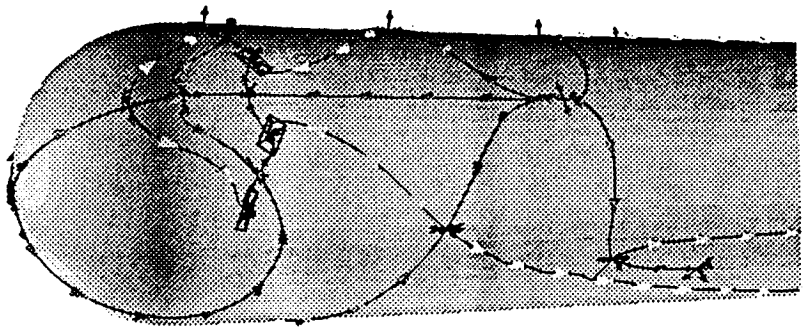

Figure 9c: TOPO's hemisphere cylinder surface flow topology.

Surface Flow Topology

By integrating curves in the two dimensional skin friction field along the eigendirections of saddles, one may visualize surface flow topology [12,21]. These curves connect skin friction field critical points and allow the user to deduce the rest of the surface skin friction field's dynamics qualitatively, since integral curves may not intersect, except at critical points. Furthermore, by ex-

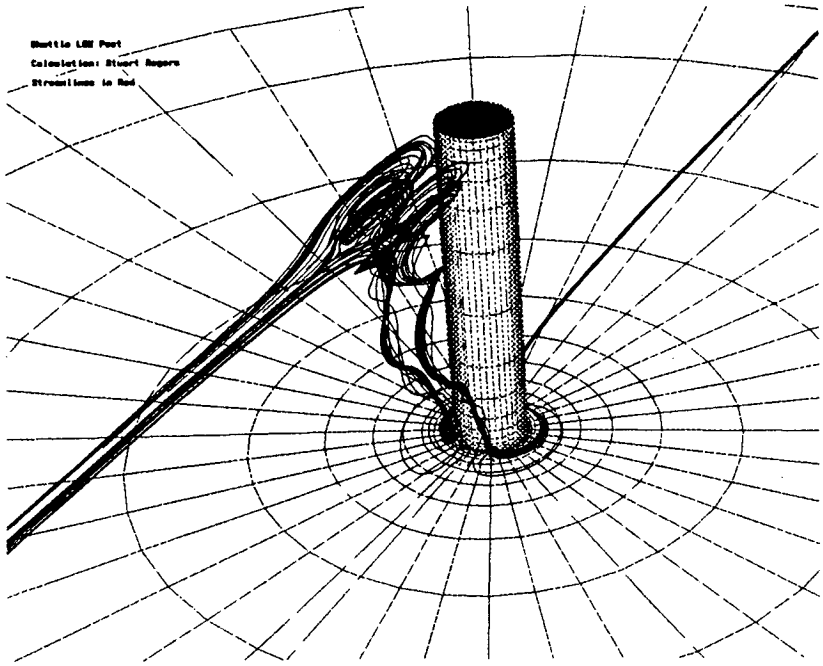

Figure 10a: Particle traces illustrating vortices.

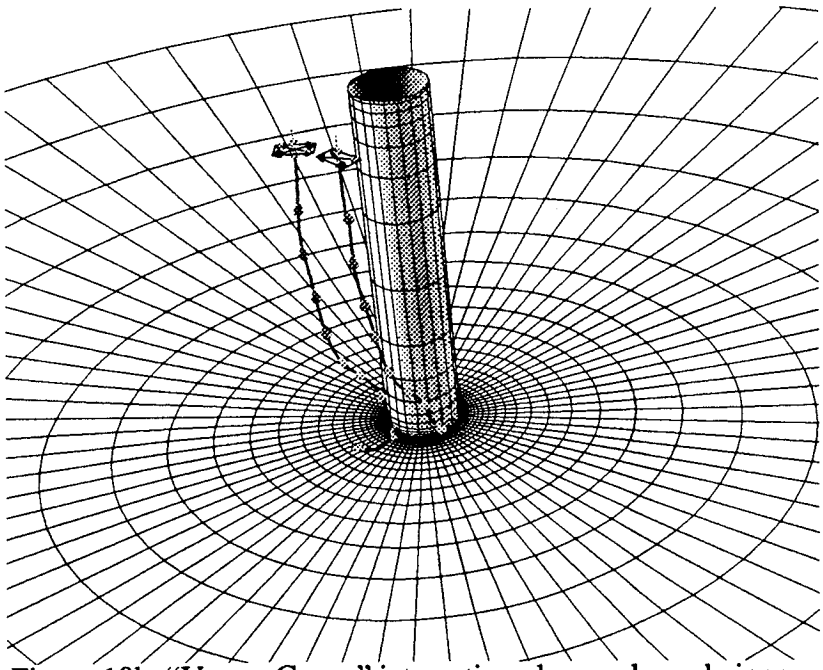

Figure 10b: "Vortex Cores:" integration along only real eigenvector of spiral-saddle critical points.

Summary

Vector fields may be visualized using the new TOPO module in FAST. A single, relatively simple picture captures many key features of **v**. The authors have used this software to visualize CFD solutions. To our knowledge, the software is unique in displaying off surface eigenvectors of skin

where both (1) and (2) have been used.

Suppose that $\frac{\partial v'_l}{\partial \xi_m}$ can be diagonalized. There exists a matrix **M** such that

$$M_{il}\frac{\partial v'_l}{\partial \xi_m}(M^{-1})_{mj} = D_{ij}$$

where **D** is diagonal. From (4) we have

$$M_{il}\frac{\partial \xi_l}{\partial x_p}\frac{\partial v_p}{\partial x_q}\frac{\partial x_q}{\partial \xi_m}(M^{-1})_{mj} = D_{ij}$$

or

$$A_{ip}\frac{\partial v_p}{\partial x_q}(A^{-1})_{qj} = D_{ij} \qquad (5)$$

where

$$A_{ip} = M_{il}\frac{\partial \xi_l}{\partial x_p} \qquad (6)$$

We conclude that

$$(\nabla \mathbf{v})_{ij} = \frac{\partial v_i}{\partial x_j} \quad \text{and} \quad (\nabla' \mathbf{v}')_{ij} = \frac{\partial v'_i}{\partial \xi_j}$$

have the same eigenvalues and that their eigenvectors are related as in (1).

Appendix B

This appendix contains the user controllable parameters of the algorithms used by TOPO.

Critical Point Search

Number of Bisections	5
<u>Newton Parameters:</u>	
Convergence criteria (epsilon)	0.000001
Minimum convergence criteria allowed	0.00001
Outer limit	1,000,000
Fuzz factor for inside-cell-check	0.000001
Maximum number of iterations	8

Curve Integration Parameters

Number of bisections	8

Santa Cruz, CA. (1984).

[12] M.J. Lighthill, "Attachment and Separation in Three Dimensional Flow," *Laminar Boundary Layers II*, ed. L. Rosenhead, pp. 72-82, Oxford University Press (1963).

[13] A.E. Perry and B.D. Fairly, "Critical Points in Flow Patterns," *Advances in Geophysics* 18 B pp. 299-315 (1974).

[14] M.S. Chong. A.E. Perry, B.J. Cantwell, "A General Classification of Three-Dimensional Flow Fields," *Phys. Fluids* A **2** (5) pp. 765-777, May (1980).

[15] A.E. Perry, "A Study of Degenerate and Non-degenerate Critical Points in Three-dimensional Flow Fields," Deutsche Forschungs- und Versuchsanstalt fur Luft und Raumsfahrt report DFVLR-FB 84-36.

[16] U. Dallman and G. Schewe, "On the Topological Changes of Separating Flow Structures at Transition Reynolds Numbers," American Institute of Aeronautics and Astronautics, paper AIAA-87-1266.

[17] U. Dallman, "Three-dimensional Vortex Structures and Vorticity Topology," *Fluid Dynamics Research* **3** pp. 183-189 (1988).

[18] A.E. Perry and J.H. Watmuff, "The Phase-Averaged Large-Scale Structures in Three-Dimensional Turbulent Wakes," *J. Fluid Mech.* **103** pp. 33-51. A.E. Perry and D.K. Tan, "Simple Three-Dimensional Vortex Motions in Coflowing Jets and Wakes," *J. Fluid Mech.* **141** pp. 197-231, (1984).

[19] S. Shirayama and K. Kuwahara, "Flow Past a Sphere: Topological Transitions of the Vorticity Field," American Institute of Aeronautics and Astronautics, paper AIAA-90-3105-CP.

[20] L. Hesselink and J. Helman, "Evaluation of Flow Topology from Numerical Data," American Institute of Aeronautics and Astronautics, paper AIAA-87-1181.

[21] J. L. Helman and L. Hesselink, "Surface Representation of Two- and Three-Dimensional Fluid Flow Topology," *Proc. Visualization '90,* San Francisco, IEEE Computer Society Press. (1990).

[22] J. L. Helman and L. Hesselink, "Analysis and Representation of Complex Structures in Separated Flows," *SPIE Conf on Extracting Meaning From Complex Data*, San Jose, (1991).

[23] J.L. Helman and L. Hesselink, "Representation and Display of Vector Field Topology in Fluid Flow Data Sets," *IEEE Computer*, pp. 27-36, Aug. 1989. Also appears in *Visualization in Scientific Computing*, G. M. Fielson & B. Shriver, eds. Companion videotape available from IEEE Computer Society Press.

[24] A. Globus, "Optree Optimization," SPIE paper 1459-01, *SPIE Conf on Extracting Meaning From Complex Data*, San Jose (1991).

[25] Mike Yamasaki, NASA Ames Research Center, personal communication.

[26] P. P. Walatka, P. G. Buning, *PLOT3D User's Manual,* NASA Technical Memorandum 101067, NASA Ames Research Center.

[27] J.A. Benek, P.G. Buning, J.L. Steger, "A 3-D Chimera Grid Embedding Technique," *AIAA 7th Computational Fluid Dynamics Conference*, Cincinnati, Ohio (1985), AIAA-85-1523.

[28] R.A. Walker, "Computing the Jordan Form for Control of Dynamic Systems," Guidance and

Hemisphere Cylinder

Calculation: Ying & Schiff

Skin Friction Topology

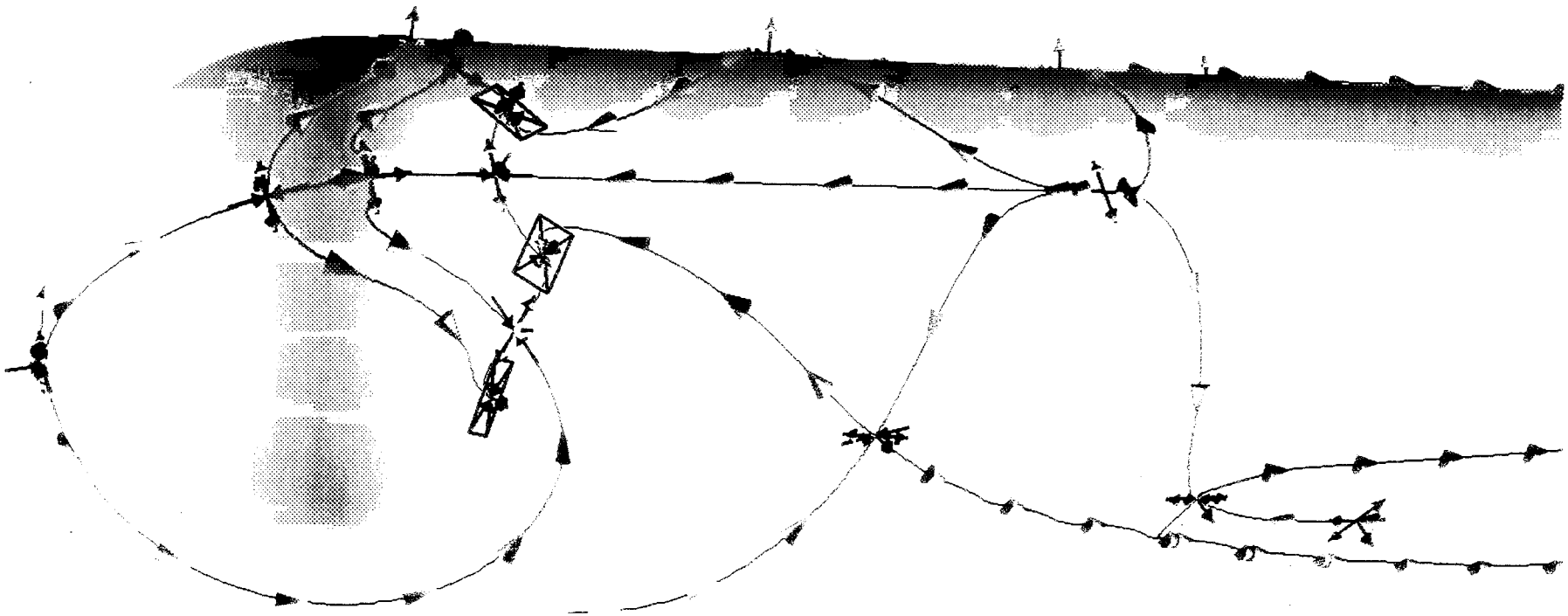

Plate 1: Hemisphere cylinder surface flow topology.

Find Interesting Portions of a Field

Data: Ying, et. al. AIAA 86-2179

Software: FAST

11 March 1991

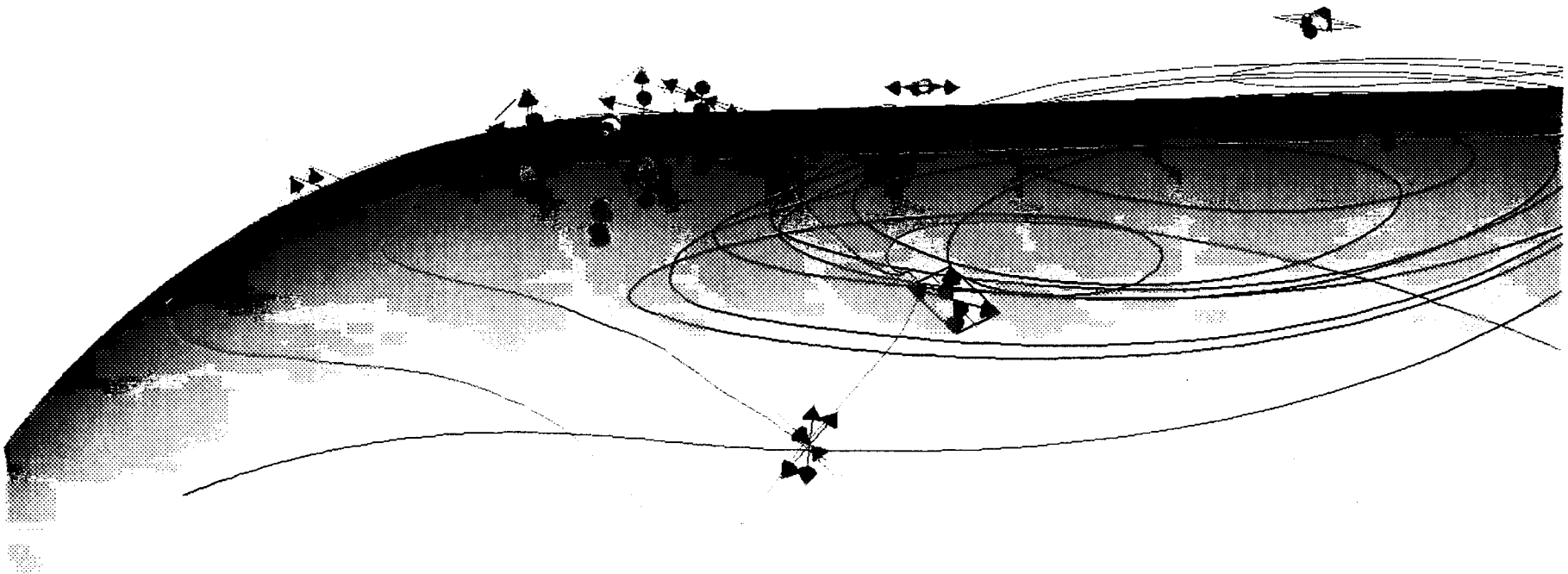

Plate 2: A particle trace (black) through an interesting portion of a hemisphere cylinder velocity field near multiple off surface critical points. Note surface flow topology in red and cyan.

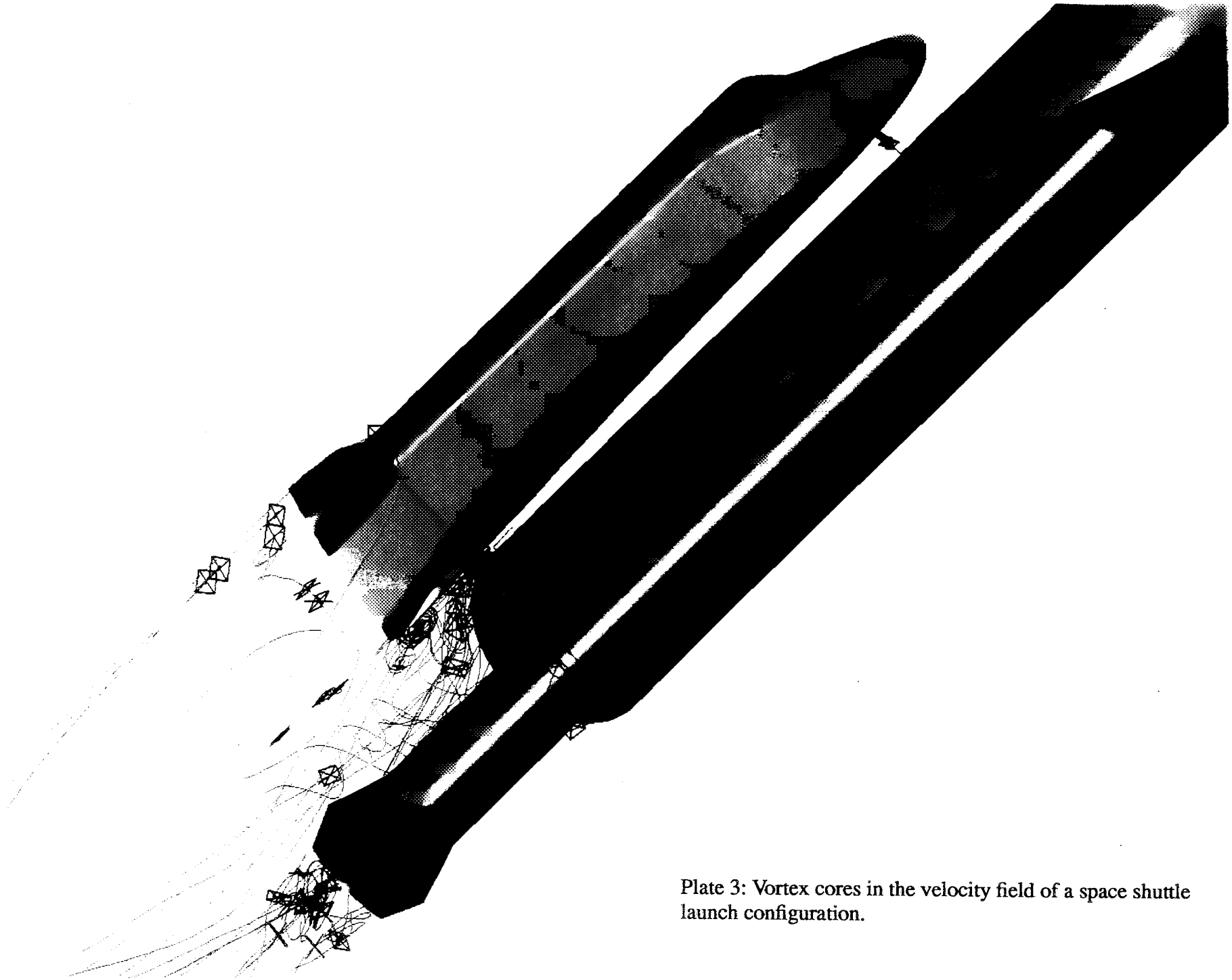

Plate 3: Vortex cores in the velocity field of a space shuttle launch configuration.